

UNITED STATES DISTRICT COURT
NORTHERN DISTRICT OF CALIFORNIA
SAN JOSE DIVISION

CISCO SYSTEMS INC,
Plaintiff,

v.

ARISTA NETWORKS, INC.,
Defendant.

Case No. [14-cv-05344-BLF](#)

**ORDER CONSTRUING CLAIMS IN
U.S. PATENT NO. 7,047,526**

[Re: ECF 91]

Plaintiff Cisco Systems Inc. brings this copyright and patent infringement lawsuit against Defendant Arista Networks, Inc. The patent portion of the lawsuit alleges Arista infringes two of Cisco's patents: U.S. Patent No. 7,047,526 (the "'526 Patent") directed at improving the control of administration and/or diagnostic software tools in processor-based systems and U.S. Patent No. 7,953,886 (the "'886 Patent") directed at providing a comprehensive extensible markup language interface for monitoring and configuring a router, while still maintaining the router's command line interface. The Court held a tutorial on March 11, 2016, and a *Markman* hearing on April 8, 2016, for the purpose construing six disputed terms in the '526 Patent and four disputes terms in the '886 Patents. Afterwards, the Patent Trial and Appeal Board instituted *inter partes* review on the '886 Patent and the Court, at Cisco's request, dismissed all claims under the '886 Patent with prejudice in lieu of the Court's indicated stay of the entire patent portion of the case.¹

¹ Arista filed a motion to strike the expert declaration of Dr. Kevin Almeroth submitted in support of Cisco's opening claim construction brief. Arista argues that Cisco failed to disclose Dr. Almeroth's opinions in accordance with the Patent Local Rules. After reviewing the briefing and Cisco's disclosures, the Court finds Cisco's disclosures were adequate. *See Reflex Packaging, Inc. v. Lenovo (U.S.), Inc.*, Case No. 10-01002-JW, 2011 WL 7295479, at *2 (N.D. Cal. Apr. 7, 2011) (holding that a plaintiff's disclosure that it intended to use the opinion of its expert that, to one of ordinary skill in the art in the field of the asserted patent, the terms at issue would have the meaning attributed to it by the Plaintiff, "support[s] a finding that Plaintiff adequately disclosed

I. BACKGROUND ON THE '526 PATENT

The '526 Patent relates to the command and interface control of administration and/or diagnostic tools for complex processor-based executable software systems. '526 Patent at 1:6-15, ECF 91-3. Typically, each administration and diagnostic tool had its own command format, function names, and syntax, which created a significant burden for system administrators. *Id.* at 1:31-37. The '526 Patent attempts to solve this issue by providing a set of universal commands that a user can use to control various administration and diagnostic tools. *Id.* at 1:58-63. By using a set of universal commands, the user only needs to learn the universal command set as opposed to learning each administration and diagnostic tool's command set. *Id.*; *see also id.* at 4:58-60. When the user inputs an universal command, a parser (software), determines which administration or diagnostic tool should be used, and translates the universal command into the appropriate format for that tool. *Id.* at 1:48-63.

A. Claim Terms at Issue

The Patent Local Rules allow the parties to identify up to 10 terms "whose construction will be most significant to the resolution of the case." Patent L.R. 4-3(c); *see also* Nortek Air Solutions v. DMG Corp., Case No. 14-cv-02919-BLF, 2015 WL 6674705, at *1, (N.D. Cal. Nov. 2, 2015) (declining to construe more than 10 terms). In contravention of the rules, the parties identified 17 terms for construction. Exhs. A and B to Joint Claim Construction Statement, ECF 70-1 and 70-2. As a result, the parties prepared an amended joint claim construction chart, which identified the following terms for construction:

1. '526 Patent

- a. "management programs";
- b. "generic command";
- c. "command parse tree";
- d. "the validating step including identifying one of the elements as a best match relative to the generic command";
- e. "the command parse tree having elements each specifying at least one

the expert testimony it intended to use."). Thus, the Court DENIES Arista's motion to strike.

corresponding generic command component and a corresponding at least one command action value”;

- f. “means for validating a generic command received from a user, the validating means configured for specifying valid generic commands relative to a prescribed generic command format and having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value, the validating means identifying one of the elements as a best match relative to the generic command”;

Am. Joint Claim Construction Chart, ECF 216.

II. LEGAL STANDARD

A. General Principles

Claim construction is a matter of law. *Markman v. Westview Instruments, Inc.*, 517 U.S. 370, 387 (1996). “It is a ‘bedrock principle’ of patent law that ‘the claims of a patent define the invention to which the patentee is entitled the right to exclude,’” *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc) (internal citation omitted), and, as such, “[t]he appropriate starting point . . . is always with the language of the asserted claim itself,” *Comark Commc’ns, Inc. v. Harris Corp.*, 156 F.3d 1182, 1186 (Fed. Cir. 1998).

Claim terms “are generally given their ordinary and customary meaning,” defined as “the meaning . . . the term would have to a person of ordinary skill in the art in question . . . as of the effective filing date of the patent application.” *Phillips*, 415 F.3d at 1313 (internal citation omitted). The court reads claims in light of the specification, which is “the single best guide to the meaning of a disputed term.” *Id.* at 1315; *see also Lighting Ballast Control LLC v. Philips Elecs. N. Am. Corp.*, 744 F.3d 1272, 1284-85 (Fed. Cir. 2014) (en banc). Furthermore, “the interpretation to be given a term can only be determined and confirmed with a full understanding of what the inventors actually invented and intended to envelop with the claim.” *Phillips*, 415 F.3d at 1316 (quoting *Renishaw PLC v. Marposs Societa’ per Azioni*, 158 F.3d 1243, 1250 (Fed. Cir. 1998)). The words of the claims must therefore be understood as the inventor used them, as such understanding is revealed by the patent and prosecution history. *Id.* The claim language,

written description, and patent prosecution history thus form the intrinsic record that is most significant when determining the proper meaning of a disputed claim limitation. *Id.* at 1315-17; *see also Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996).

Evidence external to the patent is less significant than the intrinsic record, but the court may also consider such extrinsic evidence as expert and inventor testimony, dictionaries, and learned treatises “if the court deems it helpful in determining ‘the true meaning of language used in the patent claims.’” *Philips*, 415 F.3d at 1318 (quoting *Markman*, 52 F.3d at 980). However, extrinsic evidence may not be used to contradict or change the meaning of claims “in derogation of the ‘indisputable public records consisting of the claims, the specification and the prosecution history,’ thereby undermining the public notice function of patents.” *Id.* at 1319 (quoting *Southwall Techs., Inc. v. Cardinal IG Co.*, 54 F.3d 1570, 1578 (Fed. Cir. 1995)).

B. Means-Plus-Function Claims

Paragraph 6 of 35 USC § 112 provides for means-plus-function claiming: “An element in a claim for a combination may be expressed as a means . . . for performing a specified function . . . and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.” When a claim uses the term “means” to describe a limitation, it creates a presumption that the inventor used the term to invoke § 112 ¶ 6. *Biomedino v. Waters Technologies*, 490 F.3d 946, 950 (Fed. Cir. 2007). The “presumption can be rebutted when the claim, in addition to the functional language, recites structure sufficient to perform the claimed function in its entirety.” *Id.*

If a court concludes that a claim limitation is a means-plus-function limitation, “two steps of claim construction remain: 1) the court must first identify the function of the limitation; and 2) the court must then look to the specification and identify the corresponding structure for that function.” *Id.* The claim limitation will then be construed to cover that corresponding structure and equivalents thereof. 35 USC § 112 ¶ 6.

C. Indefiniteness

“The Patent Act requires that a patent specification conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as

the invention.” *Nautilus v. Biosig Instruments*, 134 S. Ct. 2120, 2124 (2014). “A patent is invalid for indefiniteness if its claims, read in light of the specification delineating the patent, and the prosecution history, fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention.” *Id.* While the scope of the claims must be clear enough to “apprise the public of what is still open to them,” *Markman v. Westview Instruments*, 517 U.S. 370, 373 (1996), “the definiteness requirement must take into account the inherent limitations of language. Some modicum of uncertainty... is the price of ensuring the appropriate incentives for innovation.” *Nautilus*, 134 S. Ct. at 2128 (internal citations omitted). Thus, “the certainty which the law requires in patents is not greater than is reasonable, having regard to their subject-matter.” *Id.* at 2129 (quoting *Minerals Separation v. Hyde*, 242 U.S. 261, 270 (1916)).

III. CONSTRUCTION OF DISPUTED TERMS

A. “management programs”

Cisco’s Proposal	Arista’s Proposal	Court’s Construction
“separate tools or external agents having their own respective command formats that provide management functions”	“tools that are configured to execute user-entered commands having their own respective command formats rather than the generic command format”	“tools or agents configured to execute user-directed commands having their own respective command formats that provide management functions”

The disputed term “management programs” appears in independent claims 1, 10, 14, and 23 of the ’526 Patent. Claim 1 is representative of how the term is used in the claim language:

1. A method in a processor-based system configured for executing a plurality of **management programs** according to respective command formats, the method comprising:

receiving a generic command from the user;

validating the generic command based on a command parse tree that specifies valid generic commands relative to a prescribed generic command format, the command parse tree having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value, the validating step including identifying one of the elements as a best match relative to the generic command; and

issuing a prescribed command of a selected one of the **management programs** according to the corresponding command format, based on the identified one element.

1 '526 Patent at 9:19-34 (emphasis added).

2 Cisco argues that “management programs” should be construed as “separate tools or
3 external agents having their own respective command formats that provide management
4 functions.” Arista claims “management programs” should be construed as “tools that are
5 configured to execute user-entered commands having their own respective command formats
6 rather than the generic command format.” The parties’ proposed constructions differ in three
7 ways: (1) whether management programs must be separate tools or external agents; (2) whether
8 management programs may accept machine-language commands in addition to user-entered
9 commands; and (3) whether a management program’s command format can overlap with the
10 “generic command” format. The Court addresses each point of dispute in turn.

11 With respect to whether management programs must be separate tools or external agents,
12 Cisco argues the specification states that management programs may be “external agents” or
13 “external programs.” Mot. 3, ECF 91. Arista responds that the specification also states that
14 management programs may be executed “within the processor based system.” Opp. 3-4, ECF
15 141-4. Cisco replies that the use of “or” in its construction makes it clear that management
16 programs do not have to be external but can simply be separate. Reply 3, ECF 152. According to
17 Cisco, if there was no requirement that management programs be separate or external, then the
18 management programs could possibly be within the system itself. *Id.*

19 The Court agrees with Arista and finds the intrinsic evidence does not support Cisco’s
20 proposed construction. In Figure 1, the ’526 Patent discloses a system (10) with management
21 programs inside the system (18a, 18b) and external to the system (18c, 18d). ’526 Patent at 2:57-
22 3:15. While Cisco’s construction captures management programs that are external to the system,
23 it does not accurately account for the fact that the patent allows for management programs within
24 the system. Contrary to Cisco’s argument, its inclusion of the word “separate” does not account
25 for this as Cisco argues “separate” specifically excludes management programs “within the system
26 itself.” Reply 3, ECF 152. Since management programs may be either within or outside the
27 system, there is no need to limit management programs to “separate” or “external” programs.

28 With respect to whether management programs may accept machine-language commands

1 in addition to user-entered commands, at the *Markman* hearing, Arista clarified that its proposed
2 construction was not intended to exclude situations where a machine inputs commands. *Markman*
3 Tr. 56:4-9, ECF 239. Arista included “configured to execute *user-entered* commands” (emphasis
4 added) to capture that the purpose of the invention is to translate commands that would otherwise
5 be user-entered. *Id.* at 56:11-57:7. Arista’s primary concern is that Cisco will try to expand the
6 scope of this patent to cover situations where instructions are issued by a computer to another part
7 of a computer that a user never interacted with or directed. *Id.* at 63:10-24. According to Arista,
8 that would be an impermissible expansion of the patent because the invention is directed towards
9 minimizing the amount of command formats and syntax that users have to learn. ’526 Patent at
10 1:41-44.

11 The Court shares Arista’s concern that Cisco’s construction may allow for an interpretation
12 that covers situations that were never intended to be directed by a user in the first place. *Markman*
13 Tr. at 63:25-64:10. In order to avoid those situations, the Court includes “user-directed
14 commands” in the construction of management programs. The inclusion of “user-directed” does
15 not exclude embodiments disclosing inputs by computers as those embodiments disclose computer
16 inputs that were user-directed. The inclusion of “user-directed” reflects that the purpose of the
17 invention is to simplify user-directed commands.

18 Finally, with respect to whether a management program’s command format can overlap
19 with the “generic command” format, at the *Markman* hearing, Arista explained that it was not
20 seeking to prevent any overlap between the management program’s command format and “generic
21 command” format with its proposed construction. *Markman* Tr. 57:17-23. Rather, Arista was
22 seeking to show that there had to be some difference between the “generic command” format and
23 the management programs’ command format. *Id.* According to Arista, when there are a plurality
24 of management programs, if the command sets were identical, the invention’s purpose would not
25 be achieved. *Id.* at 59:23-61:1. Ultimately, Arista conceded that there was no prohibition on any
26 overlap between the command sets, and the Court will not adopt Arista’s construction with respect
27 to the overlap. In sum, the Court construes “management programs” as “tools or agents configured
28 to execute user-directed commands having their own respective command formats that provide

management functions.”

B. “generic command”

Cisco’s Proposal	Arista’s Proposal	Court’s Construction
“command that provides an abstraction of the tool-specific command formats and syntax, enabling a user to issue the command based on the relative functions, as opposed to the specific syntax for a corresponding tool”	Indefinite. or “command having a format and syntax that is an abstraction of the command formats and syntaxes of more than one management program, as opposed to the specific syntax for any such management program”	“command that provides an abstraction of the tool-specific command formats and syntax, enabling a user to issue the command based on the relative functions, as opposed to the specific syntax for a corresponding tool”

The disputed term “generic command” appears in independent claims 1, 10, 14, and 23, and dependent claims 6, 14, 15, and 19 of the ’526 Patent. Claim 1 is representative of how the term is used in the claim language:

1. A method in a processor-based system configured for executing a plurality of management programs according to respective command formats, the method comprising:

receiving a **generic command** from the user;

validating the **generic command** based on a command parse tree that specifies valid **generic commands** relative to a prescribed **generic command** format, the command parse tree having elements each specifying at least one corresponding **generic command** component and a corresponding at least one command action value, the validating step including identifying one of the elements as a best match relative to the **generic command**; and

issuing a prescribed command of a selected one of the management programs according to the corresponding command format, based on the identified one element.

’526 Patent at 9:19-34 (emphasis added).

Cisco contends that “generic command” should be construed as a “command that provides an abstraction of the tool-specific command formats and syntax, enabling a user to issue the command based on the relative functions, as opposed to the specific syntax for a corresponding tool.” Mot. 4-6, ECF 91. Arista counters that the term “generic command” is indefinite, and if it is not, it should be construed as “command having a format and syntax that is an abstraction of the

1 command formats and syntaxes of more than one management program, as opposed to the specific
2 syntax for any such management program.” Opp. 4-6, ECF 141-4.

3 First, the Court addresses whether “generic command” is indefinite. According to Arista,
4 the term is indefinite because a skilled artisan could not determine with reasonable certainty, what
5 is and is not “generic.” Opp. 5-6, ECF 141-4. As examples, Arista claims the ’526 Patent’s lead
6 inventor and Cisco’s expert could not identify whether a word is a generic command. *Id.*

7 Cisco responds that the term “generic command” is not indefinite. Cisco argues that Arista
8 misconstrues the standard for indefiniteness by isolating the term from the context of the patent.
9 Reply 3-4, ECF 152. According to Cisco, its expert explained the need for proper context to
10 determine whether certain words were generic commands, since a generic command represents an
11 abstraction. *Id.* With proper context, Cisco argues the term is not indefinite. *Id.*

12 The Court agrees with Cisco and finds that the term “generic command” is not indefinite.
13 A “generic command” is an abstraction of specific commands. As a result, with context, a skilled
14 artisan can determine with reasonable certainty whether a word represents an abstraction of a
15 specific. Contrary to Arista’s assertion, the test is not whether a skilled artisan can determine
16 whether a word in isolation is a generic command. Instead, the test is whether a skilled artisan can
17 discern the meaning of a claim term in light of the specification. *See, e.g. Ethicon Endo-Surgery,*
18 *Inc. v. Covidien, Inc.*, 796 F.3d 1312, 1338 (Fed. Cir. 2015). Here, the meaning of “generic
19 command” can be ascertained with reasonable certainty. ’526 Patent at 3:30-35.

20 Since the Court finds “generic command” is not indefinite, the Court turns to construing
21 “generic command.” Both parties derive their construction from the description of generic
22 command in the ’526 Patent specification. ’526 Patent at 3:30-35. Cisco uses the patent’s
23 description of generic command verbatim. Mot. 4, ECF 91. Arista argues that the patent’s
24 description must be modified because the patent describes a “generic command set” while the term
25 at issue is “generic command.” Opp. 7, ECF 141-4. As a result, Arista adds “command formats
26 and syntaxes of more than one management program” to its proposed construction. *Id.*

27 The Court finds Cisco’s proposed construction better construes the use of “generic
28 command” in the patent. The patent defines a generic command set as a set “that provides an

abstraction of the tool-specific command formats and syntax, enabling a user to issue the command based on the relative functions, as opposed to the specific syntax for a corresponding tool.” ’536 Patent at 3:32-35. A generic command set is simply more than one generic command. This distinction does not require the addition of the limitation proposed by Arista—that the generic command consists of command formats and syntaxes of more than one management program. Accordingly, the Court construes “generic command” as “command that provides an abstraction of the tool-specific command formats and syntax, enabling a user to issue the command based on the relative functions, as opposed to the specific syntax for a corresponding tool.”

C. “command parse tree”

Cisco’s Proposal	Arista’s Proposal	Court’s Construction
“a hierarchical data representation having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value”	<p>“tree”: “data structure consisting of linked nodes, with a root node (a node with no parent nodes), and where the remaining nodes are either a branch node (a node with a parent node and one or more children nodes), or a leaf node (a node with a parent node and no children nodes)”</p> <p>“command parse tree”: “tree for interpreting commands where each node, or element, corresponds to one or more command components”</p>	“a hierarchal data structure”

The disputed term “command parse tree” appears in independent claims 1, 10, and 14, and dependent claims 3, 11, 12, 15, and 16 of the ’526 Patent. Claim 1 is representative of how the term is used in the claim language:

1. A method in a processor-based system configured for executing a plurality of management programs according to respective command formats, the method comprising:

receiving a generic command from the user;

validating the generic command based on a **command parse tree** that specifies valid generic commands relative to a prescribed generic command format, the **command**

parse tree having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value, the validating step including identifying one of the elements as a best match relative to the generic command; and

issuing a prescribed command of a selected one of the management programs according to the corresponding command format, based on the identified one element.

'526 Patent at 9:19-34 (emphasis added).

At the *Markman* hearing, both parties agreed that “command parse tree” should be construed as “a hierarchical data structure.” *Markman* Tr. 75:25-76:6, ECF 239. Accordingly, the Court adopts this construction.

D. “the validating step including identifying one of the elements as a best match relative to the generic command”

Cisco's Proposal	Arista's Proposal	Court's Construction
Plain and ordinary meaning (except that specific terms appearing within the phrase should be construed as proposed above)	Indefinite. or “the validating step having the capability of both identifying the element in the parse tree that exactly matches the generic command, and, in the absence of an exact match, identifying the element that contains the last validated component of the generic command”	Plain and ordinary meaning (except for terms appearing within the phrase already construed by the Court)

The disputed term “the validating step including identifying one of the elements as a best match relative to the generic command” appears in independent claims 1 and 14 of the '526 Patent. Claim 1 is representative of how the term is used in the claim language:

1. A method in a processor-based system configured for executing a plurality of management programs according to respective command formats, the method comprising:

receiving a generic command from the user;

validating the generic command based on a command parse tree that specifies valid generic commands relative to a prescribed generic command format, the command parse tree having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value, **the validating step including identifying one of the elements as a best match relative to**

1 **the generic command;** and

2 issuing a prescribed command of a selected one of the management programs
3 according to the corresponding command format, based on the identified one element.

4 ’526 Patent at 9:19-34 (emphasis added).

5 Cisco claims that the term “the validating step including identifying one of the elements as
6 a best match relative to the generic command” should be given its plain and ordinary meaning.

7 Mot. 9-10, ECF 91. Arista argues that the term is indefinite because it contains the phrase
8 “generic command,” and if it is not indefinite, it should be construed as “the validating step having
9 the capability of both identifying the element in the parse tree that exactly matches the generic
10 command, and, in the absence of an exact match, identifying the element that contains the last
11 validated component of the generic command.” Opp. 9-10, ECF 141-4.

12 As explained *supra* II.B, the term “generic command” is not indefinite. Cisco argues that
13 no construction is necessary because it is used according to its plain and ordinary meaning. Mot.
14 9, ECF 91. Cisco argues that Arista’s construction improperly imports limitations from disclosed
15 embodiments to limit the scope of potential best match algorithms. Reply 5-6, ECF 152. Arista
16 counters that the term must be defined to clarify that the best match validating step must be
17 capable of handling valid and invalid commands. Opp. 9-10, ECF 141-4. Arista also argues that
18 the ’526 Patent did not invent all possible best match algorithms and the specification only
19 provides support for the specific best match algorithms disclosed in the embodiments. *Id.*

20 The claim language and intrinsic evidence supports Cisco’s proposed construction of the
21 disputed term. First, the plain and ordinary meaning of the term accurately conveys that the
22 validating step is not limited to valid commands but also includes situations involving invalid
23 commands. Second, the intrinsic evidence does not support limiting best match algorithms to the
24 embodiments disclosed in the specification. In explaining best match algorithms, the specification
25 expressly states that “it is to be understood that the invention is not limited to the disclosed
26 embodiments....” ’526 Patent 4:63-64. Arista’s construction is improperly limited to the
27 disclosed embodiments. Although the claims are read “in view of the specification, of which they are
28 a part, [the Court does] not read limitations from the embodiments in the specification into the claims.”

1 *See Hil-Rom Servs., Inc. v. Stryker Corp.*, 755 F.3d 1367, 1372 (Fed. Cir. 2014). “Even when the
2 specification describes only a single embodiment, the claims of the patent will not be read restrictively
3 unless the patentee has demonstrated a clear intention to limit the claim scope using words or
4 expressions of manifest exclusion or restriction.” *Id.* Here, Arista has not pointed to and the Court has
5 not found any clear intention to limit the claim scope to a particular embodiment disclosed in the
6 patent. It is also for this reason that Arista’s cited cases, *Netword, LLC v. Central Corp.*, 242 F.3d
7 1347 (Fed. Cir. 2011) and *On Demand Mach. Corp. v. Ingram Indus., Inc.*, 442 F.3d 1331 (Fed. Cir.
8 2006), are inapposite. In both those cases, the patentee expressed a clear intention to limit the scope of
9 the invention. *See Netword*, 242 F.3d at 1353 (specification and statements during prosecution
10 specifically defined disputed term); *Demand*, 442 F.3d at 1340 (limiting term “when the scope of the
11 invention is clearly stated in the specification, and is described as the advantage and distinction of the
12 invention”). Since best match algorithms are not limited to the disclosed embodiments, the term does
13 not need to be construed beyond its plain and ordinary meaning. Thus, the court adopts the plain and
14 ordinary meaning for the construction of “the validating step including identifying one of the elements
15 as a best match relative to the generic command,” with the caveat that any terms within this phrase
16 defined by the Court are given that meaning.

17 **E. “the command parse tree having elements each specifying at least one**
18 **corresponding generic command component and a corresponding at least one**
19 **command action value”**

Cisco’s Proposal	Arista’s Proposal	Court’s Construction
Plain and ordinary meaning (except that specific terms appearing within the phrase should be construed as proposed above)	<p>“elements”: “nodes”</p> <p>“command action value”: “piece of data that uniquely represents the prescribed command.”</p> <p>the entire phrase: “the command parse tree having nodes, such that each node specifies a unique command action value for each generic command component.”</p>	<p>“command action value”: “a value that identifies a prescribed command”</p> <p>the entire phrase: “the command parse tree having elements, such that each element specifies at least one command action value for each generic command component”</p>

28 The disputed term “the command parse tree having elements each specifying at least one

corresponding generic command component and a corresponding at least one command action value” appears in independent claims 1 and 14 of the ’526 Patent. Claim 1 is representative of how the term is used in the claim language:

1. A method in a processor-based system configured for executing a plurality of management programs according to respective command formats, the method comprising:

receiving a generic command from the user;

validating the generic command based on a command parse tree that specifies valid generic commands relative to a prescribed generic command format, **the command parse tree having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value**, the validating step including identifying one of the elements as a best match relative to the generic command; and

issuing a prescribed command of a selected one of the management programs according to the corresponding command format, based on the identified one element.

’526 Patent at 9:19-34 (emphasis added).

Cisco contends that the term “the command parse tree having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value” should be given its plain and ordinary meaning. Mot. 12, ECF 91. Cisco argues that no construction is necessary because the phrase “command parse tree” was construed by the Court and nothing in larger phrase requires construction. *Markman* Tr. 70:19-21.

At the *Markman* hearing, Arista dropped “unique” from its proposed construction, *Markman* Tr. 76:14-17. Arista argues that “elements” should be construed as “nodes,” “command action value” should be construed as “piece of data that represents the prescribed command,” and the disputed term should be construed as “the command parse tree having nodes, such that each node specifies a command action value for each generic command component.” Opp. 11-12, ECF 141-4. Arista argues that construction is necessary to prevent ambiguity as to whether the term requires that each generic command component has one command action value (one to one relationship) or whether each generic command component can have multiple corresponding command action values. *Markman* Tr. 77:12-24; Opp. 11-12, ECF 141-4. Arista also argues that “command action value” should be construed as a piece of data because command action values

1 reside in trees which are data structure. Opp. 12, ECF 141-4.

2 The Court finds that construction of the disputed phrase is necessary. First, the Court finds
3 the term “elements” does not need any further construction as its plain and ordinary meaning is
4 easily understood. Second, as to “command action value,” the specification describes a command
5 action value as a value that identifies a prescribed command. ’526 Patent at 4:31-37 (the parser
6 identifies the appropriate command based on the command action value). The Court finds that
7 Arista’s proposed construction, which uses the word “data,” introduces additional ambiguity over
8 what constitutes data. Thus, the Court construes “command action value” as “a value that
9 identifies a prescribed command.”

10 Finally, as to whether there must be a one to one relationship between the generic
11 command and command action value, the Court finds that the plain language of the term—
12 “elements each specifying at least one corresponding generic command component and a
13 corresponding at least one command action value” —indicates that each generic command
14 component can have multiple command action values. If, as Arista suggests, there must be a one
15 to one relationship between each generic component and the command action value, the term “at
16 least one” before command action value would be superfluous. Moreover, Arista’s proposed
17 construction would improperly limit the scope of the term to a disclosed embodiment in Figure 2.
18 But the patent clearly notes the invention is not limited to the disclosed embodiments. ’526 Patent
19 at 4:63-64.² In order to clarify that each generic component can have more than one command
20 action value, the Court construes the term as ““the command parse tree having elements, such that
21 each element specifies at least one command action value for each generic command component.”

22
23
24
25 ² Arista also claims that Cisco argued to the patent office during an *inter partes* review proceeding
26 that each generic command can only have one command action value. However, contrary to
27 Arista’s position, Cisco did not make such a broad argument. Rather, in explaining one
28 embodiment of the ’526 Patent, Cisco described how the embodiment depicted a generic
command with one command action value. *See* Patent Owner Preliminary Response at 7, ECF
217-1 (“Figure 2 (reproduced below with annotations) illustrates in detail *an embodiment* of the
’526 patent...”)(emphasis added).

F. “means for validating a generic command received from a user, the validating means configured for specifying valid generic commands relative to a prescribed generic command format and having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value, the validating means identifying one of the elements as a best match relative to the generic command”

Cisco’s Proposal	Arista’s Proposal	Court’s Construction
<p>Function: validating a generic command received from a user</p> <p>Structure: Parser 14 in Figure 2, which includes the command word translation table 20 and the command parse tree 22, as described in 3:36-61, and equivalents</p>	<p>Functions:</p> <p>(1) validating a generic command received from a user</p> <p>(2) specifying valid generic commands relative to a prescribed generic command format,</p> <p>(3) having elements each specifying at least one corresponding generic component and a corresponding at least one command action value, and</p> <p>(4) identifying one of the elements as a best match relative to the generic command.</p> <p>Disclosed structure: A processor executing a parser, and a corresponding memory storing a command parse tree, wherein the parser executes the algorithm of Figure 3, and wherein (1) each node of the command parse tree specifies one token and a corresponding command key; (2) the top-level nodes of the command parse tree represent all possible valid first words in the input command, second-level nodes represent all possible valid second words for each valid first word in the input command, and so on;</p>	<p>Functions:</p> <p>(1) validating a generic command received from a user</p> <p>(2) specifying valid generic commands relative to a prescribed generic command format,</p> <p>(3) having elements each specifying at least one corresponding generic component and a corresponding at least one command action value, and</p> <p>(4) identifying one of the elements as a best match relative to the generic command.</p> <p>Structure: Parser 14 in Figure 2, which includes the command word translation table 20 and the command parse tree 22, as described in 3:36-61, and equivalents. Figure 3 is an alternative embodiment.</p>

Cisco and Arista agree that this claim term is in the means-plus-function format. At the

1 *Markman* hearing, both parties agreed that defining the function as consisting of one function or
 2 multiple functions would have no meaningful impact on this case. *Markman* Tr. 111:9-14;
 3 113:11-14, ECF 239. Accordingly, the Court adopts Arista's proposal for the functions.

4 With respect to the structure, both parties stated that they would not oppose having Figure
 5 2 and Figure 3 in the structure of the term but they disputed how those figures should be included.
 6 *Id.* at 112:13-18; 114:13-15. Cisco believes Figure 3 provides an alternative embodiment of
 7 Figure 2. *Id.* at 114:17-24. Cisco argues that Figure 3 is a flow chart describing decisional logic
 8 but is not the data structure itself. Reply 9, ECF 152. According to Cisco, Arista is trying to
 9 improperly limit the scope of the means-plus-function claim to one method disclosed in an
 10 embodiment in the patent. *Id.* Arista counters that Figure 2 and Figure 3 comprise one
 11 embodiment. *Markman* Tr. 115:3-6. Arista argues that Figure 2 by itself contains no explanation
 12 about how the structure works. Opp. 16, ECF 141-4. Arista argues the explanation for Figure 2
 13 appears in Figure 3 and the accompanying description in the patent. *Id.*

14 The Court agrees with Cisco and finds that Figure 2 and its accompanying description in
 15 the specification is the relevant structure necessary for carrying out the validation function.
 16 According to the '526 Patent, Figure 2 discloses "in detail the parser...[which] includes a
 17 command word translation table 20 and a command parse tree 22...[a] is configured for validating
 18 a received generic command by comparing each input command word to the command parse tree
 19 22 to determine for the received generic command a tree element 24 identified as a best match."
 20 '526 Patent at 3:36-51. This portion of "[t]he specification...clearly links or associates [these]
 21 structure[s] to the [validating function] recited in the claim," and thus the Court should adopt
 22 Cisco's proposed structure. *Omega Eng'g, Inc. v. Raytek Corp.*, 334 F.3d 1314, 1321 (Fed. Cir.
 23 2003). Based on Cisco's comments at the *Markman* hearing, the Court includes Figure 3 as an
 24 alternative embodiment. Contrary to Arista's argument, Figure 3 is not necessary for carrying out
 25 the validation function. Arista's argument fails to recognize that Cisco is not just relying on
 26 Figure 2 but also its accompanying text in the specification which provides sufficient explanation.
 27 For example, Arista relies on *In re Aoyama*, 656 F.3d 1293 (Fed. Cir. 2011) for the proposition
 28 that when the disclosed structure is a computer programmed to carry out an algorithm, the

structure is the not the general purpose computer but rather the special purpose computer programmed to perform the disclosed algorithm. But in that case, the structure disclosed by the patent was “any working computer.” *Id.* at 1295. Here, the patent discloses more than any working computer in Figure 2 and the accompanying text. Thus, the Court adopts Cisco’s proposal for the structure and includes Figure 3 as an alternative embodiment.

IV. ORDER

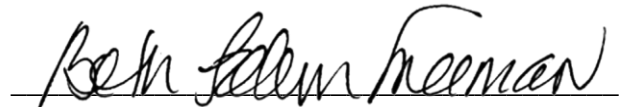
For the foregoing set forth above, the Court construes the disputed terms as follows:

Claim Term	Court’s Construction
“management programs”	“tools or agents configured to execute user-directed commands having their own respective command formats that provide management functions”
“generic command”	“command that provides an abstraction of the tool-specific command formats and syntax, enabling a user to issue the command based on the relative functions, as opposed to the specific syntax for a corresponding tool”
“command parse tree”	“a hierarchal data structure”
“the validating step including identifying one of the elements as a best match relative to the generic command”	plain and ordinary meaning (except for terms appearing within the phrase already construed by the Court)
“the command parse tree having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value”	<p>“command action value”: “a value that identifies a prescribed command”</p> <p>the entire phrase: “the command parse tree having elements, such that each element specifies at least one command action value for each generic command component”</p>
means for validating a generic command received from a user, the validating means configured for specifying valid generic commands relative to a prescribed generic command format and having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value, the validating means identifying one of the elements as a best match relative to the generic command	<p>Functions:</p> <p>(1) validating a generic command received from a user</p> <p>(2) specifying valid generic commands relative to a prescribed generic command format,</p> <p>(3) having elements each specifying at least one corresponding generic component and a corresponding at least one command action value, and</p> <p>(4) identifying one of the elements as a best match relative to the generic command.</p>

Structure: Parser 14 in Figure 2, which includes the command word translation table 20 and the command parse tree 22, as described in 3:36-61, and equivalents. Figure 3 is an alternative embodiment.

IT IS SO ORDERED.

Dated: June 15, 2016



BETH LABSON FREEMAN
United States District Judge